Machine Learning (CSE 446):
Pratical issues: optimization and learning

Sham M Kakade

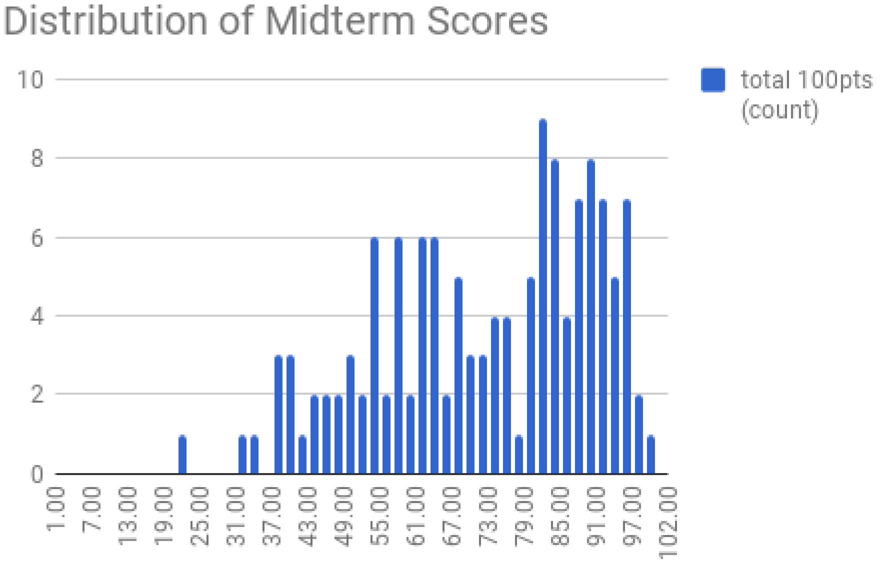
University of Washington
cse446-staff@cs.washington.edu

Announcements

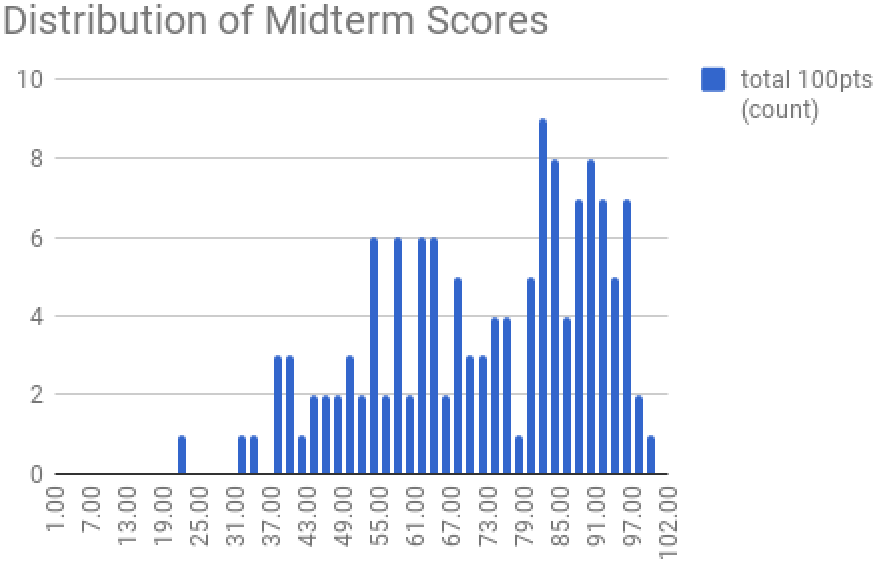
- Midterm summary:
  - stats: 71.5 std: 18
  - Office hours today: 1:15-2:30 (No office hours on Monday)
- Monday: John Thickstun guest lecture
- Grading: max (old scheme, 60% HW 15% Mid 25% Final)
- HW3 posted
  - will be periodically updated for typos/clarifictions
  - extra credit posted soon
- Today:
  - Midterm review
  - GD/SGD: practical issues

$(x_1, x_2) \to (x_1, x_2, x_1^2, x_2^2, x_1 x_2)$

Midterm



What is a good model of this distribution?



What is a good model of this distribution?

"A mixture of Gaussians"

Midterm Q4: scratch space

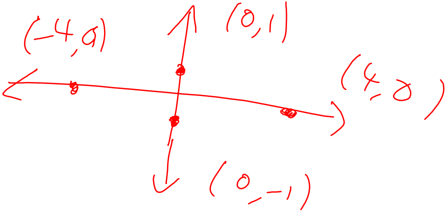
$$\underbrace{\min_w \|w\|^2}\quad \text{s.t.}\quad \underbrace{\text{for all } n,\ y_n(w\cdot x_n)\geq 1}$$

when it has a solution,

$\iff$ linearly separable

Midterm: scratch space

Midterm Q5: scratch space



$(1,0)$

$4, -4, 0, 0 \rightarrow \frac{1}{4}(4^2 + 4^2 + 0 + 0) = 8$

$(4,0), (-4,0), (0,1), (0,0)$

$\frac{1}{4}(0 + 0 + 1 + 1) = \frac{1}{2}$

Midterm: scratch space

Today

The "general" Loss Minimization Problem

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \frac{1}{N}\sum_{n=1}^{N} \underbrace{\ell(\mathbf{x}_n, y_n, \mathbf{w})}_{\ell_n(\mathbf{w})} + R(\mathbf{w})$$

How do we run GD? SGD? Which one to use?

How do run them?

Our running example

$$\underset{\mathbf{w}}{\operatorname{argmin}} \frac{1}{N} \sum_{n=1}^{N} \frac{1}{2} (y_n - \mathbf{w} \cdot \mathbf{x}_n)^2 + \frac{1}{2} \lambda \|\mathbf{w}\|^2$$

- GD? SGD?
- Note we are computing an average. What is a crude way to estimate an average?

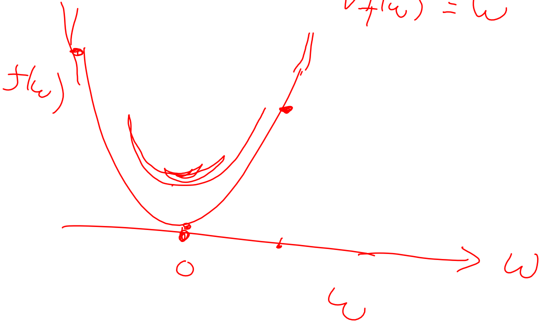
Will it converge?

How does GD behave? A 1-dim example

$$\min_{w} f(w)$$

$$f(w) = \frac{1}{2} w^2$$

$$\nabla f(w) = w$$



$$w \leftarrow w - \eta \nabla f(w)$$

$$w \leftarrow w - \eta w$$

GD: How do we set the step sizes?

- Theory:
  - square loss:
  - more generally:
- Practice:
  - square loss:
  - more generally:

try things out
to get it stable

- Do we decay the stepsize?

SGD for the square loss

Data: step sizes $\langle \eta^{(1)}, \ldots, \eta^{(K)} \rangle$
Result: parameter $\mathbf{w}$
initialize: $\mathbf{w}^{(0)} = \mathbf{0}$;
for $k \in \{1, \ldots, K\}$ **do**
    $n \sim \mathrm{Uniform}(\{1, \ldots, N\})$;
    $\mathbf{w}^{(k)} = \mathbf{w}^{(k-1)} + \eta^{(k)} \left(y_n - \mathbf{w}^{(k-1)} \cdot \mathbf{x}_n \right) \mathbf{x}_n$;
end
return $\mathbf{w}^{(K)}$;

Algorithm 1: SGD

SGD for the square loss

Data: step sizes $\langle \eta^{(1)}, \ldots, \eta^{(K)} \rangle$
Result: parameter $\mathbf{w}$
initialize: $\mathbf{w}^{(0)} = \mathbf{0}$;
for $k \in \{1, \ldots, K\}$ **do**
$\quad n \sim \mathrm{Uniform}(\{1, \ldots, N\})$;
$\quad \mathbf{w}^{(k)} = \mathbf{w}^{(k-1)} + \eta^{(k)} \left(y_n - \mathbf{w}^{(k-1)} \cdot \mathbf{x}_n \right) \mathbf{x}_n$;
end
return $\mathbf{w}^{(K)}$;

Algorithm 2: SGD

- where did the N go?
- regularization?
- minibatching?

SGD: How do we set the step sizes?

- Theory:
- Practice:
  - How do start it?
  - When do we decay it?

Stochastic Gradient Descent: Convergence

$$\mathbf{w}^* = \operatorname*{argmin}_{\mathbf{w}} \frac{1}{N} \sum_{n=1}^{N} \ell_n(\mathbf{w})$$

- $\mathbf{w}^{(k)}$: our parameter after k updates.
- Thm: Suppose $\ell(\cdot)$ is convex (and satisfies mild regularity conditions). There is decreasing sequence of step sizes $\eta^{(k)}$ so that our function value, $F(\mathbf{w}^{(k)})$, converges to the minimal function value, $F(\mathbf{w}^*)$.
- GD vs SGD: **we need to turn down our step sizes over time!**

Making features: scratch space